

程序 8-12 linux/kernel/panic.c

```
1 /*
2  * linux/kernel/panic.c
3  *
4  * (C) 1991 Linus Torvalds
5  */
6
7 /*
8  * This function is used through-out the kernel (includeinh mm and fs)
9  * to indicate a major problem.
10 */
11 /*
12  * 该函数在整个内核中使用（包括在 头文件*.h，内存管理程序 mm 和文件系统 fs 中），
13  * 用以指出主要的出错问题。
14 */
15 #include <linux/kernel.h> // 内核头文件。含有一些内核常用函数的原形定义。
16 #include <linux/sched.h> // 调度程序头文件，定义了任务结构 task_struct、初始任务 0 的数据，
17 // 还有一些有关描述符参数设置和获取的嵌入式汇编函数宏语句。
18
19 void sys_sync(void); /* it's really int */ /* 实际上是整型 int (fs/buffer.c,44) */
20
21 // 该函数用来显示内核中出现的重大错误信息，并运行文件系统同步函数，然后进入死循环--死机。
22 // 如果当前进程是任务 0 的话，还说明是交换任务出错，并且还没有运行文件系统同步函数。
23 // 函数名前的关键字 volatile 用于告诉编译器 gcc 该函数不会返回。这样可让 gcc 产生更好一些的代码，
24 // 更重要的是使用这个关键字可以避免产生某些（未初始化变量的）假警告信息。
25 // 等同于现在 gcc 的函数属性说明：void panic(const char *s) __attribute__((noreturn));
26 volatile void panic(const char * s)
27 {
28     printk("Kernel panic: %s\n\r",s);
29     if (current == task[0])
30         printk("In swapper task - not syncing\n\r");
31     else
32         sys_sync();
33     for(;;);
34 }
35
```